NEURIPS 2026

Stable GFlowNets with TV Monitoring and Probabilistic Guarantees

Zengxiang Lei[1], Ananth Shreekumar[2], Jonathan Rosenthal[2]
Ruoyu Song[2], Alvaro A. Cardenas[3], Daniel J. Fremont[3]
Dongyan Xu[2], Satish Ukkusuri[1,2*], Z. Berkay Celik[2*]

[1] Lyles School of Civil and Construction Engineering, Purdue University
[2] Department of Computer Science, Purdue University
[3] Computer Science and Engineering, University of California, Santa Cruz

Presenter: Zengxiang Lei

* Corresponding authors

GFlowNets learn a reward-proportional distribution

OBJECTIVE — SAMPLE IN PROPORTION TO REWARD

Build a terminal object x through a sequence of actions.

$$q(x) = \pi_{\text{target}}(x) = \frac{R(x)}{Z^*}$$

$$Z^* = \sum_x R(x)$$

Sample in proportion to reward while retaining diverse outcomes.

TRAINING — BALANCE FORWARD AND BACKWARD FLOWS

Forward policy P_{F} generates a trajectory:

$$\tau = (s_0 \to s_1 \to \cdots \to x)$$

Backward policy P_{B} defines reverse trajectories.

Trajectory balance:

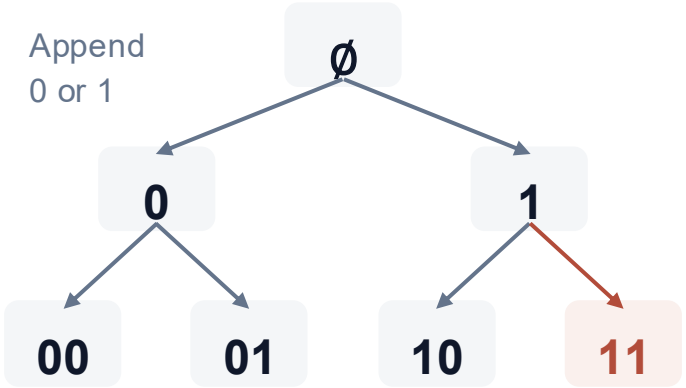
$$ZP_{\text{F}}(\tau) = R(x)P_{\text{B}}(\tau|x)$$

Learn the policies and normalizer Z by minimizing the squared log-ratio.
Model terminal distribution: $p(x) = P_{\text{T}}(x)$.

GFlowNets learn a distribution over diverse high-reward objects.

A missed sequence can create a large loss

Build a two-bit string one bit at a time. Target: $q(x) = R(x)/1000$.



x	$R(x)$	Target q	Model p
00	499	0.499	≈0.500
01	300	0.300	0.300
10	200	0.200	0.200
11	1	0.001	10^{-9}

TV measures the largest event-probability error

Total variation error: $\mathrm{TV}(p,q) = \frac{1}{2}\sum_x |p(x) - q(x)| = \max_A |p(A) - q(A)|$

TV = 0 means identical distributions; TV = 1 means disjoint support. A is any set of outcomes.

SAME EXAMPLE: PROBABILITY MASS

$$\mathbf{TV \approx 0.001}$$

At most 0.10 percentage points of error for any event A.

SAME EXAMPLE: TRAJECTORY TO 11

$$\left[\log\left(10^{-9}/0.001\right)\right]^2 \approx \mathbf{190.9}$$

Trajectory balance (TB) penalizes the log-ratio on this trajectory.

A large training loss induces excessive policy gradients, even when the model is already near optimal.

Only 00 and 11 differ. $\mathrm{TV} = 1/2\,(0.000999999 + 0.000999999)$.

A uniform loss bound controls TV

How can a nonzero training loss reflect distributional error?

LOSS-TO-TV GUARANTEE / THM 3.5

$$\mathcal{L}_{\mathrm{TB}}(\tau) \leq c^2, \forall \tau$$
$$\Rightarrow \mathrm{TV}\left(P_{\mathrm{T}}, \pi_{\mathrm{target}}\right) \leq 1 - e^{-2c}$$

Similar results hold for other GFlowNets training losses, such as Flow Matching (FM) and Detailed Balance (DB) losses.

A uniform loss bound controls TV

How can a nonzero training loss reflect distributional error?

LOSS-TO-TV GUARANTEE / THM 3.5

$$\mathcal{L}_{\mathrm{TB}}(\tau) \leq c^2, \forall \tau$$
$$\Rightarrow \mathrm{TV}\left(P_{\mathrm{T}}, \pi_{\mathrm{target}}\right) \leq 1 - e^{-2c}$$

Similar results hold for other GFlowNets training losses, such as Flow Matching (FM) and Detailed Balance (DB) losses.

But checking every trajectory is intractable.

A uniform loss bound controls TV

How can a nonzero training loss reflect distributional error?

LOSS-TO-TV GUARANTEE / THM 3.5

$$\mathcal{L}_{\mathrm{TB}}(\tau) \le c^2, \forall \tau$$
$$\Rightarrow \mathrm{TV}\left(P_{\mathrm{T}}, \pi_{\mathrm{target}}\right) \le 1 - e^{-2c}$$

Similar results hold for other GFlowNets training losses, such as Flow Matching (FM) and Detailed Balance (DB) losses.

NEXT

Replace global enumeration with finite trajectory samples.

But checking every trajectory is intractable.

From loss bounds to probabilistic guarantees

How can a nonzero training loss reflect distributional error?

LOSS-TO-TV GUARANTEE / THM 3.5

$$\mathcal{L}_{\mathrm{TB}}(\tau) \le c^2, \forall \tau$$
$$\Rightarrow \mathrm{TV}\left(P_{\mathrm{T}}, \pi_{\mathrm{target}}\right) \le 1 - e^{-2c}$$

Similar results hold for other GFlowNets training losses, such as Flow Matching (FM) and Detailed Balance (DB) losses.

FINITE-SAMPLE GUARANTEE / THM 3.6

Freeze the policy. Independently draw m backward and n forward trajectories.

$$\hat{c} = \max_{1 \le i \le m+n} \sqrt{\mathcal{L}_{\mathrm{TB}}(\tau_i)}$$

$$\mathrm{TV}\left(P_{\mathrm{T}}, \pi_{\mathrm{target}}\right) \le e^{2\hat{c}} - 1 + \log(1/\alpha)\left(\frac{1}{m} + \frac{1}{n}\right)$$

Finite-sample uncertainty is explicit.
Confidence is at least $1 - 2\alpha$.

From loss bounds to probabilistic guarantees

How can a nonzero training loss reflect distributional error?

LOSS-TO-TV GUARANTEE / THM 3.5

$$\mathcal{L}_{\mathrm{TB}}(\tau) \le c^2, \forall \tau$$
$$\Rightarrow \mathrm{TV}\left(P_{\mathrm{T}}, \pi_{\mathrm{target}}\right) \le 1 - e^{-2c}$$

Similar results hold for other GFlowNets training losses, such as Flow Matching (FM) and Detailed Balance (DB) losses.

FINITE-SAMPLE GUARANTEE / THM 3.6

Freeze the policy. Independently draw m backward and n forward trajectories.

$$\hat{c} = \max_{1 \le i \le m+n} \sqrt{\mathcal{L}_{\mathrm{TB}}(\tau_i)}$$

$$\mathrm{TV}\left(P_{\mathrm{T}}, \pi_{\mathrm{target}}\right) \le e^{2\hat{c}} - 1 + \log(1/\alpha)\left(\frac{1}{m} + \frac{1}{n}\right)$$

Finite-sample uncertainty is explicit.
Confidence is at least $1 - 2\alpha$.

Extreme trajectory losses can make these bounds loose.

Reference flow controls extreme losses

Stable GFlowNets controls extreme log-ratios by adding reference flow.

$F(\tau) = ZP_F(\tau)$: learned forward trajectory flow $R(\tau) = R(x)P_B(\tau)$: target trajectory flow δ: shared nonnegative reference flow

$$\mathcal{L}_{\text{aug}}(\tau) = \left[\log \frac{F(\tau) + \delta(\tau)}{R(\tau) + \delta(\tau)}\right]^2$$

Compute the smallest δ that keeps the augmented loss at most c^2:

$$\delta_c(\tau) = \begin{cases} \frac{ZP_F(\tau) - e^c R(\tau)}{e^c - 1} & \text{if } \log \frac{ZP_F(\tau)}{R(\tau)} > c \\ \frac{R(\tau) - e^c ZP_F(\tau)}{e^c - 1} & \text{if } \log \frac{ZP_F(\tau)}{R(\tau)} < -c \\ 0 & \text{otherwise} \end{cases}$$

SAME EXAMPLE: SEQUENCE 11

$$\boldsymbol{F = 10^{-6}, R = 1}$$

Original loss: 190.9

Add $\delta \approx 0.582$

New loss: 1 $(c = 1)$

Reference flow defines a new GFlowNet problem

1 AUGMENT FLOWS

$$F_\delta(\tau) = F(\tau) + \delta(\tau)$$
$$R_\delta(\tau) = R(\tau) + \delta(\tau)$$

→

2 EFFECTIVE TARGET

$$R_\delta(x) = \sum_{\tau \to x} R_\delta(\tau)$$
$$\pi_\delta(x) \propto R_\delta(x)$$

→

3 STANDARD GFLOWNET

$\mathcal{L}_{\text{aug}}$ is TB for π_δ with a corresponding backward policy.

The earlier loss-to-TV bounds apply to the augmented losses and target!

Reference flow guarantees

THEOREM 3.10 — DETERMINISTIC BOUND

If $\mathcal{L}_{\text{aug}}(\tau) \le c^2$ for every trajectory,

$$\mathrm{TV}\left(P_{\mathrm{T}}, \pi_{\text{target}}\right) \le (1 - e^{-2c})\left(1 + \frac{\Delta}{Z^*}\right)$$
$$= (1 - e^{-2c})\left(1 + \mathbb{E}_{\tau \sim R(\tau)/Z^*} \frac{\delta(\tau)}{R(\tau)}\right)$$

Total reference flow Δ determines the fidelity bound.

THEOREM 3.11 — FINITE-SAMPLE BOUND

Independent samples from a fixed policy; for admissible c:

$$\Pr\left[\mathrm{TV} \le \frac{e^c + (e^c - 1)M_c}{e^{-c} - (1 - e^{-c})M_c} - 1 + \varepsilon_{m,n,\alpha}\right] \ge 1 - 2\alpha$$

Observed maximum reference flow ratio M_c $= \delta_c / R$ + uncertainty.

NEXT

Turn each theorem into a computable TV tool.

Two practical TV tools

THEOREM 3.10 — DETERMINISTIC BOUND

If $\mathcal{L}_{\text{aug}}(\tau) \le c^2$ for every trajectory,

$$\text{TV} \le (1 - e^{-2c})\left(1 + \mathbb{E}_{\tau \sim R(\tau)/Z^*}\frac{\delta(\tau)}{R(\tau)}\right)$$

Total reference flow.

THEOREM 3.11 — FINITE-SAMPLE BOUND

For a fixed policy and admissible c:

$$\Pr\left[\text{TV} \le \frac{e^c + (e^c - 1)M_c}{e^{-c} - (1 - e^{-c})M_c} - 1 + \varepsilon_{m,n,\alpha}\right] \ge 1 - 2\alpha$$

Observed maximum ratio + uncertainty.

$\mathcal{M}_{\text{TV}}$ — EMPIRICAL MONITOR

Use a Monte-Carlo mean of backward sampled trajectories.
Tracks TV; no confidence guarantee.

$\mathcal{B}_{\text{TV}}$ — PROBABILISTIC CERTIFICATE

Use independent batch-level forward and backward sampled trajectories,
Confidence is at least $1 - 2\alpha$.

Full-target sampling: global TV. Restricted support: TV guarantees only for the corresponding restricted distributions.

A training loop for stable sampling

1 SAMPLE
Forward trajectories
+
reward-weighted backward trajectories

→

2 DIAGNOSE
Track extreme losses
Adapt threshold c

→

3 STABILIZE
Add minimum $\delta_c(\tau)$ and train with augmented TB

→

4 EVALUATE
$\mathcal{M}_{\mathrm{TV}}$ during training
$\mathcal{B}_{\mathrm{TV}}$ on a frozen policy

Backward sampling improves exploration; reference flow controls extremes; TV tools assess the neural sampler.

Experimental settings and evaluation protocol

1 BENCHMARKS

Tree: 3-ary, D = 5 / 10 / 15
Hypergrid: D = 4, H = 8 / 16 / 32
L14-RNA1: L = 14; 8,967 modes
sEH: $|X| \approx 10^{16}$

2 METHODS

Losses: TB [1], DB [2], FM [3]
Performance: TB, DB, SubTB [4], WDB [5], Teacher [6], Stable, StableTeacher
Ablation: TB + backward

3 TRAINING

10,000 rounds; batch 32
5 random seeds
Stable: 16 F + 16 B
StableTeacher: 16 F + 8 B + 8 T

4 EVALUATION

Hypergrid: L_1 error (10^5 final-policy samples)
RNA: discovered modes
sEH: molecular scaffolds
$\mathcal{M}_{\mathrm{TV}}$: 10 checkpoints
$\mathcal{B}_{\mathrm{TV}}$: 90% confidence

[1] Malkin et al. (2022). *Trajectory balance: Improved credit assignment in GFlowNets*. NeurIPS.
[2] Y. Bengio et al. (2023). *GFlowNet Foundations*. JMLR.
[3] E. Bengio et al. (2021). *Flow Network based Generative Models for Non-Iterative Diverse Candidate Generation*. NeurIPS.
[4] Madan et al. (2023). *Learning GFlowNets From Partial Episodes For Improved Convergence And Stability*. ICML.
[5] Silva et al. (2025). *When do GFlowNets learn the right distribution?* ICLR.
[6] Kim et al. (2025). *Adaptive teachers for amortized samplers*. ICLR.

Tree/Hypergrid: full-target backward sampling. RNA/sEH: top-10,000 cache. TV diagnostics: fixed-policy evaluation.

Reference flow improves discovery in these tasks

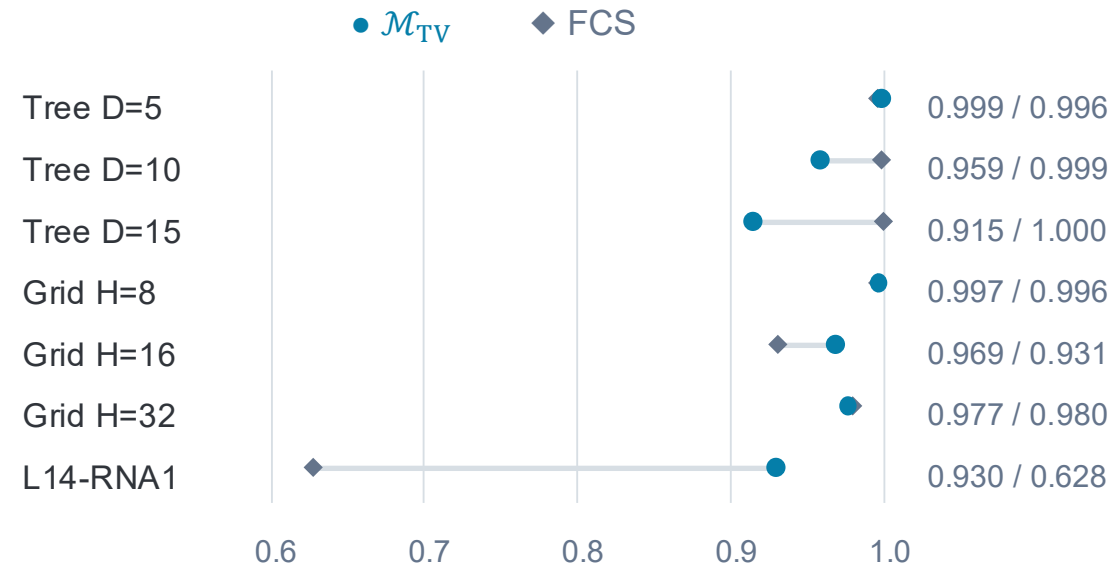
Method	Hypergrid H=32 L_1 error ↓	L14-RNA1 modes ↑	sEH scaffolds ↑
TB	1.875 ± .000	1,669 ± 300	137 ± 78
Teacher	1.603 ± .083	1,991 ± 189	6 ± 6
Stable	.713 ± .002	1,734 ± 22	**14,143 ± 2,388**
StableTeacher	.815 ± .010	**2,623 ± 131**	3,722 ± 4,188
TB + backward	.713 ± .003	1,741 ± 37	8,680 ± 2,254

Stable: +63% sEH scaffolds vs TB + backward.
StableTeacher: +32% RNA modes vs Teacher.

Mean ± SD, 5 seeds; 10,000 rounds × 32 trajectories. RNA/sEH: training discoveries; sEH raw reward ≥7.5.

$\mathcal{M}_{\mathrm{TV}}$ tracks TV without model terminal probabilities

Correlation r with exact TV across 10 checkpoints



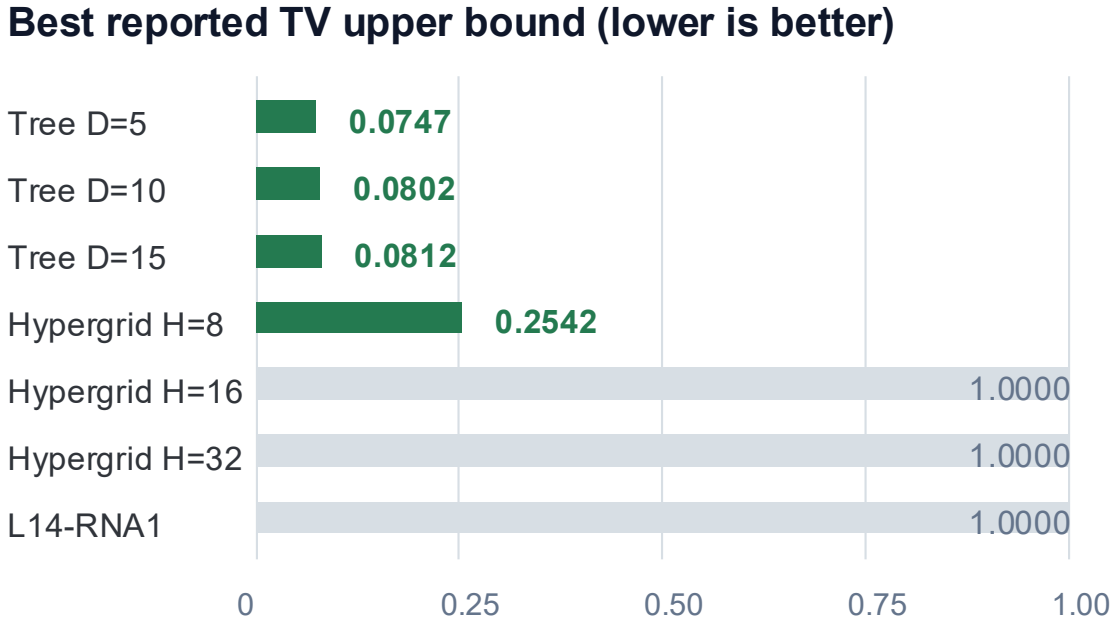
FCS = Flow Consistency in Sub-graphs [5].

$\mathcal{M}_{\mathrm{TV}}$ uses 100 backward trajectories. FCS uses the same number of trajectories with exact terminal-state probabilities here.

On L14-RNA1, $\mathcal{M}_{\mathrm{TV}}$ has $r = 0.930$ versus $r = 0.628$ for FCS.

[5] Silva et al. (2025). *When do GFlowNets learn the right distribution?* ICLR.

$\mathcal{B}_{\mathrm{TV}}$ gives formal guarantees but could be uninformative



What does 0.0747 mean?

For a preselected policy: ≤7.47 percentage points of error for any event, with 90% confidence.

A bound of 1 is uninformative. This occurs for larger Hyper grids and RNA.

Practical Implication, Stabilization, and Applications

Practical use of Stable GFlowNets

Reference flow controls extreme losses. Use $\mathcal{M}_{\mathrm{TV}}$ when model marginals are expensive. Use $\mathcal{B}_{\mathrm{TV}}$ under its stated sampling and confidence conditions.

The Stabilization & Exploration Trade-off

Active exploration (e.g., teacher outperform Stable in L14-RNA1, but combine both lead to the best performance). However, for larger scale problem like sEH, stabilization is more important for model to learn effective modes to search more effectively.

Paper: https://arxiv.org/abs/2605.01729
Code: https://github.com/tjleizeng/StableGFlowNets